

# Directions for Artificial Intelligence

Dustin Smith

## Contents

<b>Belief 1: Knowledge exists only to solve problems</b>	<b>3</b>
Conclusions and comments . . . . .	3
<b>Belief 2: Knowledge needs many representations</b>	<b>4</b>
Conclusions and comments . . . . .	5
<b>Belief 2.a Reasoners need many kinds of inference</b>	<b>5</b>
Conclusions and comments . . . . .	5
<b>Belief 3: One can only learn what they can represent</b>	<b>5</b>
Conclusions and comments . . . . .	6
<b>Belief 4: An AI should support natural language understanding and generation</b>	<b>6</b>
Conclusions and comments . . . . .	7
<b>Where do we go from here?</b>	<b>7</b>
Thinking about a solution . . . . .	7
Previous Solutions . . . . .	7
Commonsense Approach . . . . .	7
Learning by reading: . . . . .	7
Computational linguistics . . . . .	8
<b>Supplementary Materials</b>	<b>8</b>
List of Kinds of Learning . . . . .	8
Source of instruction: . . . . .	8
Examples of different kinds of lessons . . . . .	9
Examples of different learning subjects . . . . .	9
Inductive Bias . . . . .	10

In this page, I list and defend some important high-level constraints for designing cognitive architectures / AI systems. My goal is to draw connections between deep ideas from many

fragmented sub-fields in AI with the hope of showing how they can reunify toward the original, common goal.

These ideas, some more developed than others, build upon each other and were developed after reviewing work from many fields and from numerous discussions with colleagues and mentors.

## Belief 1: Knowledge exists only to solve problems

- **Knowledge is the end-product of learning, and learning does not occur arbitrarily...**
- **Biases are needed for learning to take place.**
  - When learning or inference results in multiple explanations, how do you select one? This is the problem of *inductive bias*. Also see the no free lunch theorem.
  - These biases depend on the resource constraints of the agent. The **memory-inference tradeoff**: consider the differences in how an AI with 20GB memory could learn the multiplication table for all 3 digit numbers (e.g., by simply memorizing a  $10^8$  table), versus a human, for whom memorization is more costly, who would represent equivalent knowledge by learning a *process* that could compute the solution digit by digit— albeit much more slowly than table-lookup. On top of storage required for the procedural description, the human then only needs to learn the outcomes for ways to multiply all 10x10 single digits—100 pieces of information— or 50 rather, if the learner exploits a symmetry, the communicative property:  $A \times B = B \times A$ .
- **Some impetus is required to engage learning:** See my list for many examples of different kinds of learning for some thoughts about what could activate a learning procedure.
- **Having knowledge is not enough; one needs to retrieve it in *relevant* situations.** The more you know, the harder it is to filter irrelevant knowledge; precision/recall trade off.

## Conclusions and comments

There are two major points that fall out of this belief:

1. **We need to address the problem of *machine learning within the context of problem solving*.** The work in AI's **planning** does this; “reinforcement learning” deals with the problem of *learning* plans by interacting with the world — but usually only with simple kind of learning - a value representing a reward signal.
2. **Knowledge should not be considered in isolation of process:** It is not enough to construct ontologies and taxonomies to organize knowledge without considering *what* that knowledge is going to be used for or *how* it could be automatically learned and retrieved when relevant to the agent’s goals. The whole structure of an ontology would likely collapse if the agent changes its goals.

Fortunately, it seems many others share this belief. A whole zoo of representations exist in artificial intelligence that represent *problem solving procedures*; the best of them<sup>1</sup> attempt to encapsulate semantic knowledge in a kind of modular, reusable way that generalizes to more than a single plan.

From an AI point of view, these procedures are called **plans**: e.g., STRIPS (logical *if-do-then* rules), scripts (cached STRIPS plans), production rules (e.g. ACT-R), transition matrices (MDP, HMM). There are many problems with these representations [Harold Fox's PhD thesis 2008]: either they are not general enough to be reused, or there are too many and it is hard to retrieve the appropriate actions for a particular problem.

In terms of representing *process*, there's an obvious but more general parallel in computer science, both from a practical perspective (programming languages!) and theoretical (functional/denotational/operational formalizations of programming languages). Computer programming languages, and the study of them, are much more expressive than the plan representations typically used in AI research.

## Belief 2: Knowledge needs many representations

As Gerry Sussman noted, we shouldn't be searching for a silver bullet to represent or solve all problems. This happens all too often in AI, when a researcher becomes fatally attracted to a single, simple framework (Bayesian Nets, Logic Programming, Neural Networks, ...) and tries to use it to represent and solve all problems.

A simple example of using multiple representations is the use of *instances* and *abstract classes*, often to facilitate generalization. For example, to represent three different symbols `tweety`, `bird`, `animal`, along with their taxonomic inheritance relationships `isa(tweety, bird)` and `isa(bird, animal)`. This allows properties to be affixed to different levels of detail, in order to more compactly represent information and expedite learning.

- modularizing representations for sharing structures across domains (representationally, multiple-inheritance; procedurally: overloading)
- different representations have different constraints: connection constraints, influence constraints (representationally, these are properties of relationships: associative, communicative, reflexive...)
- meta-knowledge is important for:
  - compressing knowledge (an abstract rule applies to many instances)
  - specifying overhypotheses / feature selection / learning biases

---

<sup>1</sup>Relevant approaches have been developed in the seemingly unrelated area of **sequential data mining**.

- getting unstuck in problem solving by switching through parallel representations
- Possible (maximally general) framework for thinking about representation: the relational representation with numeric weights (probabilistic logic)

### Conclusions and comments

There is not much work on this problem. Is there some underlying “substrate” we can use to talk about “all representations”? One possibility is the relational representation of statistical relational learning theory [deRaedt 2008; deRaedt and Kersting 2008], as long as we don’t confuse it with its commitments to logical inference, because ...

### Belief 2.a Reasoners need many kinds of inference

From the point of view of the meta-layer, **process is knowledge<sup>2</sup>**, so this is really not a separate point from belief 2 (e.g., learning negative expertise/ cognitive sensors)

- Again, AI researchers need to avoid searching for a single bullet (e.g., using logic *deduction* as the only inference technique)
- Can we find a (maximally general) framework for thinking about inference? graph matching / statistical relational learning?

### Conclusions and comments

Graph matching, like the concept of Analogy, may be too general an idea to be very useful, although it covers most (all?) kinds of inference: deduction, sets of variable (nodes) distance metrics, etc. Computationally, this is NP-hard.

### Belief 3: One can only learn what they can represent

- Fodor’s concept learning problem: how do you learn label ‘dog’ for concept DOG unless you have DOG representation already.
- Solution: composition + abstraction (representational example: object orienting programming: building complex structures out of smaller units and then hiding their parts through a single object symbol)
- Question: what determines the compositional hypothesis space: representationally, do we use only boolean operators, or any kind of relational property? what controls the

<sup>2</sup>For an example of meta-reasoning, consider the interaction between a *program as source code* and a *compiler*: the compiler here is the meta-reasoner, reasoning about the instructions that will later be executed on the computer.

features involved in generation? some compositional constraints are needed to control search space.

- Another problem: is there a bottom layer to this? (problem of symbol grounding) Clearly the granularity of the bottom-layer is arbitrary (depends on sensors and actuators); yet, for learning-while-problem-solving it may be important for this layer to be consistent, distinct, and immutable.

### Conclusions and comments

This problem highlights importance of good teaching: giving examples in the right order; good analogies; building concepts off of background knowledge; etc.

This problem is challenging, because with the previous beliefs (1 and 2), there are many kinds of learning going on at each stage: do you repair the plan, a part of the plan, or the process that selected the plan (or rejected others)? Do we use the idea from cognitive science that we need to learn semantic representations (e.g., taxonomies) along with procedures simultaneously? Can think of the components of plans as *if-do-then* rules (three part tuples: t1, action, t2)? If so, this could be generalization, parameterization, or composition. What about many layers? Are control statements (procedurals programming) mixed with the declarative, or always at the meta-layer?

### Belief 4: An AI should support natural language understanding and generation

- Natural Language is primary representation people use to communicate
  - articulation: an information-theoretic generation of symbols to divide a search path through a series of planning decisions?
  - to explain an idea, you need to know the contextual dependencies (you do not know the term ‘jockey’ unless you know the term ‘horse’; a concept in ‘analysis’ unless you know ‘calculus’) which is useful for both teaching and communication.

**Problems:** not all English (or language X) is the same (try reading old English; a subject you don’t understand; or, try to decipher text written by someone with incoherent or badly communicated ideas). The language-knowledge mapping and articulation/interpretation processes are likely different for every individual! Consequently, multi-author corpus-based approaches may be **unable to reveal anything about what is important to model the individual reader’s semantic knowledge.**

## Conclusions and comments

Interestingly, recent work in linguistics and psycholinguistics [7, 5, 1, 3] supports the idea that sentences are associated with Event Structures (e.g., a step of a plan), so this mapping may be an opportunity for interdisciplinary unification.

## Where do we go from here?

### Thinking about a solution

How do we evaluate a cognitive architecture, where the learning task it to learn by:

- experience: interacting with environment, learning by observation, learning from a teacher
- thinking: generating and testing plans through internal simulations

A logical low-cost environment is text, where the quest is to predict and fill in missing information in some of the text. The inference problem is restricted to the agent's goals, and the text contains a sequence of events — thus they can be associated with the agent's plans or representations of other's (possibly inanimate object's) plans.

### Previous Solutions

Previous approaches and their problems:

### Commonsense Approach

- examples: Cyc, Open Mind applications. . . .
- problems: does not generalize to new domains, knowledge is domain specific, collection is biased toward problems; even if your domain was 'all domains'- eg CYC, the problem is underspecified and inference becomes intractable.

### Learning by reading:

- examples: Charniak 76, Mitchell 2008
- Problems: The inferences one makes about a given text depend on the questions that are asked, which in turn depends on the content of the text and the relationship to the readers' goals.

## Computational linguistics

- extracting models from corpora. This has been done at the semantic frame level (e.g, Propbank) but it is questionable because:
  - most sentences were composed by different authors, who may share (or not share) the same lexicon but express or desire to communicate different concepts.
  - there is a *many to many* mapping between subcategorization frames and semantic frames. Learning this from a noisy corpus won't work.

## Supplementary Materials

### List of Kinds of Learning

Here are several examples of *kinds of learning*, based on locus of teacher (reward from environment; external teachers; from observation; from daydreaming/planning) and scale of effect (habitation/associative learning all the way through goal reformulation)

### Source of instruction:

1. The agent itself, interacting with its mind
  - discovering a solution to a hard problem
  - thinking that you do not have a skill required to perform a task, and giving up task
2. The agent interacting with the environment
  - body babbling to notice relation between “move-arms” signals and change in visual field (arm moves).
  - eating food from a truck and getting sick.
3. An external person:
  - See a person try to open a locked door (to learn that a door is locked)
  - See a person (in a movie) decide to save 10000 strangers over his own child (learning about the hero's goal priorities)
  - A stranger yells at you for walking in front of their car
  - A parent praises you for an action
  - An advisor tells you to avoid a certain research area
  - An imagined god tells you what you are doing is wrong
  - An imagined parent tells you what you are doing is good
  - You see a colleague ask a certain kind of question in a situation, and you try to figure out how they made the association.

- A student makes an improper connection between two topics, and you learn that the topic you are teaching is confusing to new learners.
- A paper tells you that the topology of a graph of the Internet resembles a jellyfish.

#### 4. From “society”

- Imagined ridicule for violating social taboos (e.g., being naked in public)
- Imagined praise for huge social work (e.g., discovering cure for cancer/AIDS)

#### **Examples of different kinds of lessons**

- Realizing that you have “found a solution” to a hard problem.
- Confirmation of success after having changed a lightbulb; installed a screen door; . . .
- Feeling full after a good dinner
- A parent physically punishes you
- A stranger praises you, but you think praise is unwarranted
- Colleague insults you — draws attention to a private problem you were aware of
- Someone insults you — draws attention to a problem you were not previously aware of
- Person on the street insults you, but you take it in stride.
- You hear second-hand of another person’s praise.
- You feel a sudden pang of pain in your chest
- You realize you are enjoying a gentle breeze pass over your body

#### **Examples of different learning subjects**

- Learning to avoid an action
- Learning to avoid a goal
- Learning a situation in which to avoid an action
- Learning a situation in which to avoid a goal
- Learning action had a new side effect
- Learning the state of an unknown (person tries to open locked door)

- Learning the correct state from a wrong previously known state
- Learning that one goal precludes another
- Learning that you are bad at solving a certain kind of problem
- Learning that you are good at solving a . . . .
- Learning that you have lost an ability
- Learning that you have gained a new ability

These can further be classified by scales of effect: e.g. from a short-term memory of a phone number to “never, ever think about this topic again!”

### Inductive Bias

An **inductive bias** is something that allows a learner to favor a particular hypothesis when multiple hypotheses fit the data. This problem arises often in machine learning, when an interpretation/hypothesis space is very large and there are many competing explanations.

For example, if you see the string “10110”, you likely interpreted it as a single number in either base-2 or base-10, but the number could have possibly come from an infinite base- $n$  hypothesis space. Why base-2 and not, say, base-4? A recent view from cognitive science [2] is that hypothesis spaces are *generated* using causal representations, whereby the relevant causal associations and inference increases to meet the complexity of the training data, thereby dynamically expanding the hypothesis set. This view naturally embodies an Occam’s razor/Minimum description length principle of spatial (representation size) and procedural (inference depth) compression. This view can be used to explain why people have difficulty discriminating random from non-random strings [6], namely, it is easier (or more likely) for a causal explanation of a random sequence (“HHHHT” is viewed as a non-random) than a truly random underlying process.

An inductive bias can be considered anything that puts an ordering on the hypothesis space, and there are many stages in the learning process where this can happen. Following [4] there are at least three kinds of inductive biases: a language bias, search bias and validation bias.

### References

- [1] GLENBERG, A., JAWORSKI, B., RISCHAL, M., AND LEVIN, J. What brains are for: Action, meaning, and reading comprehension. *Reading Comprehension Strategies: Theories* (Jan 2007).

- [2] KEMP, C., AND TENENBAUM, J. B. Theory-based induction. *Proceedings of the 25th Annual Conference of the Cognitive ...* (Jan 2003).
- [3] LEVIN, B. Objecthood: An event structure perspective. *Proceedings of CLS* (Jan 1999).
- [4] NEDELLEC, C., ROUVEIROL, C., ADE, H., BERGADANO, F., AND TAUSEND, B. Declarative bias in ilp. *Advances in inductive logic programming* 32 (1996), 82–103.
- [5] TAYLOR, L., LEV-ARI, S., AND ZWAAN, R. A. Inferences about action engage action systems. *Brain and Language* (Jan 2007).
- [6] WILLIAMS, J., AND GRIFFITHS, T. Why are people bad at detecting randomness? because it is hard. *cocosci.berkeley.edu*.
- [7] ZWAAN, R. A., AND TAYLOR, L. J. Seeing, acting, understanding: motor resonance in language comprehension. *Journal of experimental psychology General* 135, 1 (Feb 2006), 1–11.